Respectfully submitted,

Richard C. Auchterlonie
Reg. No. 30,607
ARNOLD WHITE & DURKEE
P.O. Box 4433
Houston, Texas 77210
(713) 787-1400

Date: _6 Sept. 2001_

APPENDIX I. Version with Markings to Show Changes Made.

In the Specification:

Page 3 lines 1 to 12:

Unfortunately, data consistency problems may arise if concurrent client access to a read/write file is permitted through more than one data mover. These data consistency problems can be solved in a number of ways. For example, as described in Vahalia et al., U.S. Patent [_____] 5,893,140 issued [_____] April 6, 1999 [[Serial No. 08/747,631 filed Nov. 13, 1996]], entitled "File Server Having a File System Cache and Protocol for Truly Safe Asynchronous Writes," incorporated herein by reference, locking information can be stored in the cached disk array, or cached in the data mover computers if a cache coherency scheme is used to maintain consistent locking data in the caches of the data mover computers. However, as shown in FIG. 1, labeled "Prior Art," a more elegant solution to the data consistency problem has been implemented at EMC Corporation in a network file server system having multiple stream server computers and one or more cached disk arrays.

Page 3, lines 13 to 27:

FIG. 1 shows a network file server system having at least two data mover computers 21 and 22. The first data mover 21 has exclusive access to read/write files in a first file system 23, and the second data mover 22 has exclusive access to read/write files in a second file system 24. As shown, the file systems [12, 14] 23, 24 are respective volumes of data contained in the same cached disk array 25, although alternatively each file system [12, 14] 23, 24 could be contained

in a respective one of two separate cached disk arrays. For example, each of the data mover computers 21, 22 has a respective high-speed data link to a respective port of the cached disk array 25. The cached disk array 25 is configured so that the file system 23 is accessible only through the data port connected to the first data mover 21 and so that the file system 24 is accessible only through the data port connected to second data mover 22. Each of the data movers 21, 22 maintains a directory of the data mover ownership of all of the files in the first and second file systems 23, 24. In other words, each of the data movers maintains a copy of the file system configuration information in order to recognize which data mover in the system has exclusive access to a specified read/write file.

Page 11, line 22 to page 12 line 1:

Referring to FIG. 2, there is shown a network file server 40 that may provide a significant improvement in data access time by using a data bypass path around the data mover that owns the file system during the transmission of read/write data. The network file server includes at least two data movers 41, 42 that access at least two file systems 43, 44 in storage of a cached disk array 45. The first data mover 41 owns the file system 43, and the second data mover 42 owns the second file system 44. The file server [30] 40 is linked by a data network 50 to a plurality of clients 46, 47. The first data mover 41 has a network port 51 for receiving file access requests from at least one client 46, and the second data mover 42 has a network port 52 for receiving file access requests from at least one other client 47.

Page 13 lines 8 to 21:

If a write operation changes any of the file attributes, then the new file attributes are written from the first data mover 41 to the second data mover, and after the write data is committed to the second file system 44, the second data mover 42 commits any new file attributes by writing the new file attributes to the file system. As described in the above-referenced Vahalia et al., U.S. Patent [_____] 5,893,140 issued [_____] April 6, 1999 [[Serial No. 08/747,631 filed Nov. 13, 1996]], a data security problem is avoided by writing any new file attributes to storage after the data are written to storage. If the network communication protocol supports asynchronous writes, it is possible for a data mover that does not own a file system to cache read or write data, but in this case any data written to the cache should be written down to the nonvolatile storage of the file system and the cache invalidated just prior to releasing the lock upon the file system. Otherwise, data in the cache of a data mover that does not own a file system may become inconsistent with current data in the file system or in a cache of another data mover.